Multivariate Versus Univariate Kriging Metamodels for Multi-Response Simulation Models (Revision of 2012-039)

Kleijnen, Jack P.C.; Mehdad, E.

Publication date:
2014

No. 2014-012

MULTIVARIATE VERSUS UNIVARIATE KRIGING METAMODELS FOR MULTI-RESPONSE SIMULATION MODELS

By

Jack P.C. Kleijnen, Ehsan Mehdad

14 February, 2014

This is a revised version of CentER Discussion Paper

No. 2012-039
16 May, 2012

ISSN 0924-7815
ISSN 2213-9532

Multivariate versus univariate Kriging metamodels for multi-response simulation models

Jack P.C. Kleijnen, Ehsan Mehdad
Tilburg School of Economics and Management, Tilburg University
Postbox 90153, 5000 LE Tilburg, The Netherlands

Abstract

To analyze the input/output behavior of simulation models with multiple responses, we may apply either univariate or multivariate Kriging (Gaussian process) metamodels. In multivariate Kriging we face a major problem: the covariance matrix of all responses should remain positive-definite; we therefore use the recently proposed "non-separable dependence" model. To evaluate the performance of univariate and multivariate Kriging, we perform several Monte Carlo experiments that simulate Gaussian processes. These Monte Carlo results suggest that the simpler univariate Kriging gives smaller mean square error.

Keywords: Simulation, Stochastic processes, Multivariate statistics
JEL: C0, C1, C9, C15, C44

1 Introduction

In operations research (OR) practice, *simulation* is often applied. Simulation may be either *deterministic* or *random* (stochastic). Applications of deterministic simulation abound in engineering such as computer aided engineering (CAE), but there are also applications in OR as demonstrated by the following two examples. Example 1 concerns the management of fisheries at the French Research Institute for Exploitation of the Sea (IFREMER);

see Mahevas & Pelletier (2004). Example 2 is the case study on the CO_2 greenhouse effect by Kleijnen et al. (1992). Applications of random simulation are plentiful in OR, especially in queueing and inventory management; see the references in Kleijnen (2008, pp. 3-6).

Kriging model may be used to analyze the input/output (I/O) behavior of a given simulation model; this analysis may serve validation, sensitivity analysis, and optimization, as discussed by Kleijnen (2008). This Kriging gives a *metamodel*; i.e., it approximates the I/O function defined by the underlying simulation model. There are different types of metamodels; most popular is a polynomial of either first or second order; see Kleijnen (2008). We, however, focus on Kriging, which has already become popular in engineering and is gaining popularity in OR; see the many references in Chen et al. (2012) and Kleijnen (2008). Most of this Kriging literature, however, ignores multivariate Kriging; also see our literature summary below.

In practice, a given simulation model has *multiple* outputs—also called responses or performance criteria. For example, Kleijnen (1993) discusses a case study on the production planning of steel tubes of different types, using a simulation model with 28 outputs which—after a discussion with management—were reduced to two outputs. Kleijnen & Smits (2003) discusses multiple performance metrics in supply chain management. The literature on metamodels, however, often reduces these multiple outputs to a single output—either ignoring all the other outputs or combining all outputs through a weighting function; in our Monte Carlo experiments (detailed In Section 4) we shall briefly discuss results for the sum and the product of two outputs. Other publications present metamodels per *individual* output ignoring the correlations between outputs; e.g., Kleijnen et al. (2010) fit univariate Kriging models for each of the two outputs—namely, cost and service—of a call-center simulation. In all our Monte Carlo experiments we also apply such univariate Kriging—besides multivariate Kriging.

Intuitively, it may seem that multivariate Kriging gives a lower *mean squared error*(MSE) than univariate Kriging, because the former accounts for the cross-correlations between different output types, whereas the latter accounts only for the auto-correlations between outputs of the same type for different input combinations—as we shall explain in Sections 2 and 3. However we think this intuition may be misleading. In practice the Kriging parameters are unknown so they must be estimated, which increases the MSE; multivariate Kriging requires the estimation of additional parameters—namely, the cross-correlations—which further increases the MSE. Note that Hernan-

dez & Grover (2013) also use the MSE criterion in their article on Kriging.

To empirically compare univariate and multivariate Kriging, we use *Monte Carlo experiments* that guarantee the validity of the Kriging metamodel. The literature usually experiments with realistic simulation models, but these experiments imply approximation errors (bias) of the Kriging metamodels. Moreover, these simulation models may be computationally expensive. We limit our investigation to Kriging in deterministic simulation, which is also the basis for Kriging in stochastic simulation.

Furthermore, we limit our first Monte Carlo experiments to situations with a single input and two outputs. Many OR problems do have a single input; examples are queueing simulation models with the traffic rate as the single input and inventory models such as the newsvendor problem with the order quantity as the single input. Moreover, Kriging in simulation usually assumes that in case of multiple inputs the correlation function is the product of the correlation functions per individual input; see equation (2). In this example we limit the number of multiple outputs to two; in case of more outputs, the cross-correlations are correlations between all pairs of outputs. We do vary the magnitudes of the cross-correlation between the two outputs. In the second example we base our Monte Carlo experiment on a climate simulation with five inputs and three outputs.

To provide some *background* for our study, we summarize the rather limited number of publications that explicitly discuss multiple outputs. This literature assumes *different types* of multivariate models; we distinguish the following three types:

1. In practice, simulation models may have (say) n types of output; each type is a specific transformation of the same input combination and the same pseudorandom number stream; in deterministic simulation, this stream vanishes. Software (such as Arena) for building and running discrete-event simulation models permits the automatic collection of multiple outputs. Not only simulation may give multiple outputs; real-life systems may too. This type is the focus of our study.

2. A given real system may be represented by n different simulation models with different degrees of realism (detail); so-called multi-fidelity simulation. We claim that this situation is extremely rare in OR. The simulation model with few details is run for many input combinations, whereas the detailed type is run for fewer combinations. Obviously, the most detailed simulation is the real system itself. See Santner et al.

(2003); Forrester et al. (2008); Goh et al. (2013); Tuo et al. (2013), and also "partially heterotropic" situations in Wackernagel (2003, p. 158).

3. Besides the output of interest, the modelers collect information on the gradient of this output. In discrete-event simulation, this type is rare, because the estimation of this gradient is complicated (it typically uses either "perturbation analysis" or the "score function" method). An example is Chen et al. (2013). Obviously, the output and its gradient are estimated for the same input combinations.

For type-1 real-life systems, Cressie (1991, pp. 138-142) speaks of *cokriging* in his book on spatial data analysis. Wackernagel (2003, pp. 143-209) also discusses geostatistics, so he restricts the input data to one, two, or three dimensions (whereas simulation implies an arbitrary number of dimensions). Gneiting et al. (2010) also discuss cokriging in geostatistics assuming so-called Matérn correlation functions. Santner et al. (2003, pp. 101-116) do discuss simulation or computer experiments, assume type-3 simulations. Higdon et al. (2008) discuss the combination of real-life "field data" and simulation data, where both types of data concern the same real-life system so it concerns type-2 situations; they allow for very many types of output. Forrester (2010) also discusses type-2 situations; i.e., the combination of (i) scarce and expensive real-life data with abundant and inexpensive simulation data, or (ii) scarce and expensive data from a detailed simulation model with abundant and inexpensive data from a quick-and-dirty simulation model. Williams et al. (2010) discuss multivariate Kriging in constrained optimization in simulation with multiple outputs—but they follow Santner et al. (2003). Altogether we recommend Santner et al. (2003) and Wackernagel (2003) for an introduction to multivariate Kriging. Note that Li et al. (2006) also recognize that in practice simulation models have multiple outputs and that Kriging is an important type of metamodel, but those authors use a completely different approach (they do not use cokriging with estimated cross-correlations).

Besides the areas of operations research (our focus), geostatistics, and engineering there is another area with major contributions to Kriging or Gaussian process (GP); namely *machine learning*; see Rasmussen & Williams (2005). Multivariate GPs are investigated in machine learning in multi-task learning, multi-sensor networks or structured output data. To obtain positive definite (PD) covariance matrixes, this community uses either so-called separable models or nonseparable models. The nonseparable models are based on

either convolution method or the linear model of coregionalization (LMC). We define these different models in Section 3. Separable models for multi-task learning are used by Bonilla et al. (2007). Álvarez et al. (2011) show how several models in machine learning are special cases of LMC. In LMC, they use Cholesky's decomposition of the cross-covariance matrix to construct a PD covariance matrix, and they show that the convolution method gives lower standardized mean square errors than LMC. Fricker et al. (2010) present an LMC variant that uses eigendecomposition of the cross-covariance matrix to construct a PD covariance matrix. They show that their LMC variant gives a lower mean squared error than the convolution method. The convolution method is introduced to this community by Boyle & Frean (2005). The main disadvantage of this method are the computational and storage requirements. Álvarez & Lawrence (2011) propose a more efficient approximation for multivariate GPs constructed through the convolution method. This method does not spend much effort on accurate modeling of cross-covariance. To improve the accuracy in convolution method, more parameters are needed; Fricker et al. (2010) propose a new method that introduces such parameters. Note that Constantinescu & Anitescu (2013) specify the covariance matrixes imposing constraints originating from the physics laws that determine relationships among the outputs of their application; in OR, however, such knowledge is usually not available.

We summarize our article as follows. We consider multivariate Kriging model constructed through LMC which is proposed by Fricker et al. (2010). Furthermore, we interpret this novel Kriging model. We also present Monte Carlo results for the performance of this multivariate model and univariate Kriging per output. Using this Monte Carlo laboratory, we confirm previous results showing that multivariate Kriging does not provide improvements compared with univariate Kriging—even under *ideal* conditions. Svenson & Santner (2010) use Fricker et al. (2010)'s LMC for their multi-objective optimization problem; unlike we, they do not compare univariate and multivariate Kriging. Fricker et al. (2010) find that univariate Kriging always gives smaller RMSEs than multivariate Kriging. Fricker et al. (2010) suggest that if the output is a function of other outputs, then multivariate Kriging outperform univariate Kriging. We use the data in Fricker et al. (2010) only to select the parameters in our Monte Carlo experiment with a multivariate Kriging metamodel that has no specification errors; i.e., their Kriging metamodel is only an approximation of the true I/O function of their underlying simulation model, whereas multivariate Kriging in our Monte Carlo labo-

ratory gives a metamodel without any bias. So, instead of selecting some arbitrary data that might accidentally favor or "bias" our methodology, we base our experiments on Fricker et al. (2010)'s data. Note that in the appendix we give details, including several statistical tests for verifying the correctness of Monte Carlo experiments with Kriging; such tests are necessary because computer codes may contain unintended programming errors and peers should be enabled to reproduce results.

We organize the remainder of this article as follows. Section 2 summarizes the basics of univariate Kriging, including references to computational issues. Section 3 extends this Kriging to multivariate Kriging with nonseparable dependence structure, including technical details. Section 4 describes our Monte Carlo laboratory with GP models so the Kriging assumptions are guaranteed and we can use this laboratory to empirically compare univariate and multivariate Kriging. Section 5 presents conclusions and topics for future research. The references at the end of this article enable the reader to study more aspects of this challenging topic.

2 Basic univariate Kriging

The various disciplines that apply Kriging, use different terminologies. We have already observed that geostatisticians speak of "sites", whereas simulationists speak of "points" or "combinations". In machine learning, the "old" points are called the "training set". Simulationists use correlation functions, whereas geostatisticians use the related concept of variograms.

Our notation remains close to the notation in DACE—the free univariate MATLAB Kriging toolbox developed and well-documented by Lophaven et al. (2002), assuming deterministic simulation. For the reader's convenience, Appendix A includes Table A.1, summarizing the symbols used. Note that alternative free software is mentioned by Frazier (2010), Kleijnen (2008, p. 146), and Roustant et al. (2012). Commercial software called JMP is offered by SAS. Several authors present a Bayesian interpretation of the Kriging model, but we follow a frequentist approach.

Suppose the given simulation model is run for m *combinations* of the k simulation inputs $\mathbf{x} = (x_1, \ldots, x_k)^\top$. These combinations are also called "locations" or "scenarios"; Lophaven et al. (2002) call them "sites", which stems from the origin of Kriging; namely, geostatistics (Daniel Krige was a mining engineer in South Africa). Simulating these m input combinations

gives the outputs $\mathbf{y} = (y_1, \ldots, y_m)^\top$.

Like most authors on Kriging in simulation, we assume *Ordinary Kriging*:

$$y = \mu + z \tag{1}$$

where μ denotes the mean output, and z a *stationary GP* with zero mean. Because z is stationary, z has a constant—but unknown—variance σ_z^2, and its covariances $c_{i;i'}$ $(i, i' = 1, \ldots, m)$ between the outputs of the input combinations $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ are determined by the *distance* between $\mathbf{x}_i = (x_{i;1}, \ldots, x_{i;k})^\top$ and $\mathbf{x}_{i'}$ in the k-dimensional input space (we use the symbol $c_{i;i'}$ instead of $\sigma_{i;i'}$ because in multivariate Kriging $\sigma_{g;g'}$ refers to the covariances between the outputs of type g and g'). Kriging of simulation models with their possibly high-dimensional input space assumes that these covariances are the *products* of the k individual correlation functions; e.g., a so-called *Gaussian correlation function* implies

$$c_{i;i'} = \sigma_z^2 \Pi_{j=1}^k \exp[-\theta_j (x_{j;i} - x_{j;i'})^2] \tag{2}$$

where $\theta_j \geq 0$ measures the importance of input j; $x_{j;i}$ is the i^{th} entry of the j^{th} simulation input; and $|x_{j;i} - x_{j;i'}|$ measures the distance in the input dimension j between the combinations i and $i^{'}$. Note that if $\theta_j = 0$, then changes in input j have no effect at all on the covariance $c_{i;i'}$. If $\theta_j = \infty$, then the covariance $c_{i;i'}$ reduces to zero, so the outputs at the locations i and i' are independent. The covariances $c_{i;i'}$ are gathered in the symmetric and positive-definite $m \times m$ covariance matrix $\mathbf{\Sigma}$, and the corresponding correlations $c_{i;i'}/\sigma_z^2$ are collected in $\mathbf{R}$ so $\mathbf{\Sigma} = \sigma_z^2 \mathbf{R}$. The two extreme values for the correlation coefficient ($\theta_j = 0$ or $\theta_j = \infty$) give a singular covariance matrix, because this matrix has identical columns. Note that Simpson et al. (2001) claim that Ordinary Kriging defined in (1) with a Gaussian correlation function defined in (2) is the most common Kriging model in engineering.

The classic Kriging predictor assumes *known (hyper)parameters* μ and $\mathbf{\Sigma}$. Requiring the predictor to be linear and unbiased and using the mean squared prediction error (MSPE) criterion, the *best linear unbiased predictor* (BLUP) for the output y_0 of $\mathbf{x}_0$ is

$$\widehat{y_0} = \mu + \mathbf{c}_0{}^\top \mathbf{\Sigma}^{-1} (\mathbf{y} - \mu \mathbf{1}) \tag{3}$$

where $\mathbf{1}$ denotes the m-dimensional vector with ones; $\mathbf{c}_0 = (c_{0;1}, \ldots, c_{0;m})^\top$ the vector with the covariances between the outputs at the new and the m old

input combinations (so $\mathbf{\Sigma} = (\mathbf{c}_1, \ldots, \mathbf{c}_m)$ with vectors $\mathbf{c}_i = (c_{i;1}, \ldots, c_{i;m})^\top$); and $(\mathbf{y}-\mu\mathbf{1})$ the vector with residuals. Note that $\mathbf{x}_0$ denotes the "new" input combination; an alternative notation replaces the subscript 0 by $m+1$. If $\mathbf{x}_0$ is actually one of the old points $\mathbf{x}_i$ $(i = 1, ..., m)$, then the predictor $\widehat{y_i}$ equals the observed output y_i; i.e., Kriging gives an exact interpolator.

In practice, however, the parameters μ and $\mathbf{\Sigma}$ are unknown, so a major problem is their *estimation*. Note that Σ includes the variance σ_z^2 on its main diagonal and the k parameters θ_j assuming the Gaussian correlation function (2). Santner et al. (2003) use *maximum likelihood estimation* (MLE). Because z in (1) follows a *GP*, the density function (say) f of $\mathbf{y}$ is

$$f(\mathbf{y}) = \frac{1}{(2\pi)^{m/2}(|\mathbf{\Sigma}|)^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{y}-\mu\mathbf{1})^\top \mathbf{\Sigma}^{-1}(\mathbf{y}-\mu\mathbf{1})\right] \tag{4}$$

where $|\mathbf{\Sigma}|$ denotes the determinant of $\mathbf{\Sigma}$. This density function is denoted by $\mathcal{N}_m(\boldsymbol{\mu}, \mathbf{\Sigma})$ with $\boldsymbol{\mu} = \mu\mathbf{1}$.

MLE minimizes the *log-likelihood function* $l(\mathbf{\Sigma},\mu|\mathbf{y})$, while ignoring terms that do not depend on the parameters μ and $\mathbf{\Sigma}$ to be estimated; i.e., (4) implies

$$l(\mathbf{\Sigma},\mu|\mathbf{y}) = \ln|\mathbf{\Sigma}| + (\mathbf{y}-\mu\mathbf{1})^\top \mathbf{\Sigma}^{-1}(\mathbf{y}-\mu\mathbf{1}). \tag{5}$$

The resulting MLE estimators are denoted by $\widehat{\mathbf{\Sigma}}$ and $\widehat{\mu}$. This minimization is a difficult mathematical problem. The classic solution in Kriging is to "divide and conquer" through the application of mathematical statistics, as follows.

We use $\mathbf{\Sigma} = \sigma_z^2\mathbf{R}$ to replace $|\mathbf{\Sigma}|$ by $|\mathbf{R}|\,(\sigma_z^2)^m$ and $\mathbf{\Sigma}^{-1}$ by $\mathbf{R}^{-1}/\sigma_z^2$, and obtain

$$l(\mathbf{R},\mu|\mathbf{y}) = m\ln\sigma_z^2 + \ln|\mathbf{R}| + \frac{(\mathbf{y}-\mu\mathbf{1})^\top \mathbf{R}^{-1}(\mathbf{y}-\mu\mathbf{1})}{\sigma_z^2}. \tag{6}$$

Following Santner et al. (2003) and also Gano et al. (2006), we minimize this function in the following steps:

1. Initialize; i.e., select preliminary values for $\widehat{\boldsymbol{\theta}} = (\widehat{\theta_1}, \ldots, \widehat{\theta_k})^\top$ which together define $\widehat{\mathbf{R}}$.

2. Compute the generalized least squares (GLS) estimator of the mean:

$$\widehat{\mu} = (\mathbf{1}^\top \widehat{\mathbf{R}}^{-1}\mathbf{1})^{-1}\mathbf{1}^\top \widehat{\mathbf{R}}^{-1}\mathbf{y}. \tag{7}$$

3. Substitute $\widehat{\mu}$ resulting from Step 2 and $\widehat{\mathbf{R}}$ resulting from Step 1 into the variance estimate

$$\widehat{\sigma_z^2} = \frac{(\mathbf{y}-\widehat{\mu}\mathbf{1})^\top \widehat{\mathbf{R}}^{-1}(\mathbf{y}-\widehat{\mu}\mathbf{1})}{m}. \tag{8}$$

Note that $\widehat{\sigma_z^2}$ uses the denominator m, whereas the classic unbiased estimator assuming $\mathbf{R} = \mathbf{I}$ would use $m - 1$.

4. Solve the remaining problem in (6):

$$\min_{\widehat{\boldsymbol{\theta}}} \left[m \ln \widehat{\sigma_z^2} + \ln \left| \widehat{\mathbf{R}} \right| \right]. \tag{9}$$

5. Use the $\widehat{\boldsymbol{\theta}}$ that solves (9) to update $\widehat{\mathbf{R}}$, and substitute the resulting $\widehat{\mathbf{R}}$ into (7) and (8).

These estimated Kriging parameters result in the estimated MSPE or variance of the Kriging predictor (3):

$$\widehat{\text{MSPE}} = \widehat{\sigma_z^2} + \left(\mathbf{1}^\top \widehat{\boldsymbol{\Sigma}}^{-1} \widehat{\mathbf{c}}_0 \right)^\top \left(\mathbf{1}^\top \widehat{\boldsymbol{\Sigma}}^{-1} \mathbf{1} \right)^{-1} \left(\mathbf{1}^\top \widehat{\boldsymbol{\Sigma}}^{-1} \widehat{\mathbf{c}}_0 \right) - \widehat{\mathbf{c}}_0{}^\top \widehat{\boldsymbol{\Sigma}}^{-1} \widehat{\mathbf{c}}_0. \tag{10}$$

Minimization problem defined in (9) is difficult because of "the multimodal and long near-optimal ridge properties of the likelihood function" ; i.e., this problem is not convex; see Gano et al. (2006), Jones et al. (1998, p. 486), and Marrel et al. (2010, p. 5). The problem of a flat likelihood function leading to highly variable MLE is tackled by Li & Sudjianto (2005), adding a penalty function to the likelihood function.

3 Multivariate Kriging

In this section we consider $n \geq 1$ outputs for each of the m input combinations (type-1 model in Section 1); i.e., the simulation outputs become $y_{i;g}$ $(i = 1, \ldots, m)$ $(g = 1, \ldots, n)$. For the reader's convenience, we collect symbols in the Appendixes A and B including Tables A.1 and B.1. In multivariate Kriging, we can still use the *multinormal* density defined for the univariate case in (4)—provided we define the stacked vector (say) $\mathbf{Y}$ with mn elements such that we first gather the n outputs at the first input combination $\mathbf{y}_1 = (y_{1;1}, \ldots, y_{1;n})^\top$ (first row of Table B.1), then the n outputs at the second input combination $\mathbf{y}_2 = (y_{2;1}, \ldots, y_{2;n})^\top$, etc., until finally the n outputs at the m^{th} input combination $\mathbf{y}_m = (y_{m;1}, \ldots, y_{m;n})^\top$ (last row of Table B.1). Note that y_g (output of type g with $g = 1, \ldots, n$) has the constant mean μ_g (see Table A.1). The resulting vector $\mathbf{Y}$ has the multivariate normal density function $\mathcal{N}_{mn}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_\mathbf{Y})$ where $\boldsymbol{\mu}$ denotes the mean vector with mn elements and $\boldsymbol{\Sigma}_\mathbf{Y}$ denotes the $mn \times mn$ covariance matrix of $\mathbf{Y}$:

$$f(\mathbf{Y}) = \frac{1}{(2\pi)^{mn/2} (|\boldsymbol{\Sigma}_\mathbf{Y}|)^{1/2}} \exp \left[-\frac{1}{2} (\mathbf{Y} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}_\mathbf{Y}^{-1} (\mathbf{Y} - \boldsymbol{\mu}) \right]. \tag{11}$$

For $\mathbf{\Sigma_Y}$ we offer the following comments.

In this section we discuss the general case with k inputs and n outputs (in Appendix C we detail the simplest multivariate Kriging example; namely, $k = 1$ input and $n = 2$ outputs). A stationary GP for each type of output implies that the output y_g has the constant variance σ_g^2 and (auto)covariances that decrease with the distance between its input combinations; again see (2). Moreover, in multivariate Kriging different output types $y_g(\mathbf{x}_i)$ and $y_{g'}(\mathbf{x}_{i'})$ with $g; g' = 1, \ldots, n$ and $g \neq g'$ have cross-covariances, when simulated for the same or for different input combinations; i.e., $\mathbf{x}_i$ may be the same as $\mathbf{x}_{i'}$ or may be different. For example, if $n = 2$ (bivariate output), then

$$\mathrm{Cov}(\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})) = \begin{bmatrix} \mathrm{cov}(y_1(\mathbf{x}_i), y_1(\mathbf{x}_{i'})) & \mathrm{cov}(y_1(\mathbf{x}_i), y_2(\mathbf{x}_{i'})) \\ \mathrm{cov}(y_1(\mathbf{x}_i), y_2(\mathbf{x}_{i'})) & \mathrm{cov}(y_2(\mathbf{x}_i), y_2(\mathbf{x}_{i'})) \end{bmatrix}.$$

If we have $\mathbf{x}_i = \mathbf{x}_{i'}$ in this example with $n = 2$, then the 2×2 matrix (say) $\mathbf{\Sigma}_0$ does not vary with the input combination $\mathbf{x}$ and becomes

$$\mathbf{\Sigma}_0 = \begin{bmatrix} \sigma_1^2 & \sigma_{1;2} \\ \sigma_{1;2} & \sigma_2^2 \end{bmatrix} \tag{12}$$

where $\sigma_{1;2} = \mathrm{cov}(y_1, y_2)$. In the general case with n outputs, the (symmetric) covariance matrix at input combination i ($i = 1, \ldots, m$) is

$$\mathbf{\Sigma}_0 = \begin{bmatrix} \sigma_1^2 & \sigma_{1;2} & \ldots & \sigma_{1;n} \\ & \sigma_2^2 & \ldots & \sigma_{2;n} \\ & & \ddots & \vdots \\ & & & \sigma_n^2 \end{bmatrix}. \tag{13}$$

So in the general case, $\mathbf{Y}$ has the $mn \times mn$ covariance matrix

$$\mathbf{\Sigma_Y} = \begin{bmatrix} \mathbf{\Sigma}_0 & \mathrm{Cov}(\mathbf{Y}(\mathbf{x}_1), \mathbf{Y}(\mathbf{x}_2)) & \ldots & \mathrm{Cov}(\mathbf{Y}(\mathbf{x}_1), \mathbf{Y}(\mathbf{x}_m)) \\ \mathrm{Cov}(\mathbf{Y}(\mathbf{x}_1), \mathbf{Y}(\mathbf{x}_2)) & \mathbf{\Sigma}_0 & \ldots & \mathrm{Cov}(\mathbf{Y}(\mathbf{x}_2), \mathbf{Y}(\mathbf{x}_m)) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{Cov}(\mathbf{Y}(\mathbf{x}_1), \mathbf{Y}(\mathbf{x}_m)) & \mathrm{Cov}(\mathbf{Y}(\mathbf{x}_2), \mathbf{Y}(\mathbf{x}_m)) & \ldots & \mathbf{\Sigma}_0 \end{bmatrix}. \tag{14}$$

To *predict* the n outputs at input combination $\mathbf{x}_0$, we define—analogously to $\mathbf{c}_0$ defined below (3)—the $n \times mn$ matrix

$$\mathbf{\Sigma}_{0;m;n} = (\mathrm{Cov}(\mathbf{Y}(\mathbf{x}_0), \mathbf{Y}(\mathbf{x}_1)), \ldots, \mathrm{Cov}(\mathbf{Y}(\mathbf{x}_0), \mathbf{Y}(\mathbf{x}_m)))$$

and obtain—analogously to (3)—the multivariate BLUP

$$\widehat{\mathbf{y}}(\mathbf{x}_0) = \widehat{\boldsymbol{\mu}} + \boldsymbol{\Sigma}_{0;m;n}\boldsymbol{\Sigma}_{\mathbf{Y}}^{-1}(\mathbf{Y} - \mathbf{F}\widehat{\boldsymbol{\mu}}) \tag{15}$$

where the vector $\widehat{\boldsymbol{\mu}}$—analogously to (7)—denotes the GLS estimator

$$\widehat{\boldsymbol{\mu}} = (\mathbf{F}^\top\boldsymbol{\Sigma}_{\mathbf{Y}}^{-1}\mathbf{F})^{-1}\mathbf{F}^\top\boldsymbol{\Sigma}_{\mathbf{Y}}^{-1}\mathbf{Y} \tag{16}$$

with $\mathbf{F} = \mathbf{1}_m \otimes \mathbf{I}_n$ where $\mathbf{1}_m$ denotes an m-dimensional vector with ones, $\otimes$ the Kronecker operator, and $\mathbf{I}_n$ the $n \times n$ unity matrix; also see Svenson & Santner (2010).

The estimated MSPE of the multivariate Kriging predictor (15) is

$$\widehat{\mathrm{MSPE}}[\widehat{\mathbf{y}}(\mathbf{x}_0)] = \widehat{\boldsymbol{\Sigma}_0} - \widehat{\boldsymbol{\Sigma}_{0;m;n}}(\widehat{\boldsymbol{\Sigma}_{\mathbf{Y}}})^{-1}\widehat{\boldsymbol{\Sigma}_{0;m;n}}^\top + \mathbf{U}\left(\mathbf{F}^\top(\widehat{\boldsymbol{\Sigma}_{\mathbf{Y}}})^{-1}\mathbf{F}\right)^{-1}\mathbf{U}^\top \tag{17}$$

with $\mathbf{U} = \mathbf{I}_n - \widehat{\boldsymbol{\Sigma}_{0;m;n}}(\widehat{\boldsymbol{\Sigma}_{\mathbf{Y}}})^{-1}\mathbf{F}$.

Now we consider one particular way to obtain a $\boldsymbol{\Sigma}_{\mathbf{Y}}$ that is PD. Our formalization follows the *nonseparable dependence model* in Svenson & Santner (2010), who in turn follow Fricker et al. (2010), who precede Fricker et al. (2013). Fricker et al. (2010) explain both nonseparable models and separable models. Separable models assume $\boldsymbol{\Sigma}_{\mathbf{Y}} = \boldsymbol{\Sigma}_0\mathbf{R}$ where we defined the cross-covariance matrix $\boldsymbol{\Sigma}_0$ in (13) and the auto-covariance matrix $\mathbf{R}$ below (2); i.e., separable models assume that the matrix $\boldsymbol{\Sigma}_{\mathbf{Y}}$ can be separated into two components with the second component implying that all outputs have the same auto-correlation matrix. Moreover, Fricker et al. (2010) show that separable models have an undesirable so-called Markov property; see equation (6) in Fricker et al. (2010) or equation (4) in Fricker et al. (2013). An example of a separable model is Zhang (2007). Furthermore Fricker et al. (2010, 2013) discuss how nonseparable models may be created through either convolution method or LMC. These former methods convolve a Gaussian white noise process with a smoothing kernel; see Ver Hoef & Barry (1998); Higdon (2002). Moreover, Fricker et al. (2010) present empirical results that suggest that convolution method gives worse results than LMC, so we limit our research to LMC. Originally, geostatistics uses LMC to model spatial multivariate processes, see (Wackernagel, 2003, pp. 194-200). In LMC the output process is a linear combination of building-block processes. Fricker et al. (2010) use an LMC with n blocks. Here we detail LMC following Fricker et al. (2010).

Obviously, an n-variate Gaussian variable with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ may be generated from a vector $\mathbf{Z}$ with n normally independently identically distributed (NIID) "standard" variables (which have zero

means and unit variances) through $\boldsymbol{\mu} + \boldsymbol{AZ}$ with $\boldsymbol{\Sigma} = \mathbf{A}\mathbf{A}^\top$ where $\mathbf{A}$ is a symmetric matrix. Svenson & Santner (2010) extend this idea, and consider

$$\boldsymbol{Y} = \boldsymbol{\mu} + \boldsymbol{AZ} \tag{18}$$

where $\mathbf{Y}$ denotes the n-variate output at any input combination, $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)^\top$ is the vector of GP means, $\mathbf{A} = (a_{g;g'})$ is a symmetric and PD matrix, and $\mathbf{Z}$ is a vector of mutually independent stationary GPs with zero mean and unit variance. More precisely, $\mathbf{Z}$ has the Gaussian correlation function defined in (2). It is simple to derive that (18)—together with $\mathbf{R}$ defined below (2) and defining $\boldsymbol{\theta}^{(l)} = (\theta_1^{(l)}, \ldots, \theta_k^{(l)})^\top$ with $l = 1$, ..., n—implies

$$\mathrm{Cov}(\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})) = \mathbf{A}\mathrm{diag}[\mathbf{R}(\mathbf{x}_i - \mathbf{x}_{i'}; \boldsymbol{\theta}^{(1)}), \ldots, \mathbf{R}(\mathbf{x}_i - \mathbf{x}_{i'}; \boldsymbol{\theta}^{(n)})]\mathbf{A}^\top. \tag{19}$$

We point out that we stack the covariance matrixes per input combination , not per output. If $\mathbf{x}_i = \mathbf{x}_{i'}$, then (2) gives $\exp[-\theta_j(x_{j;i} - x_{j;i'})^2] = 1$ so (19) implies that $\boldsymbol{\Sigma}_0$—defined in (13)—becomes

$$\mathrm{Cov}(\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_i)) = \boldsymbol{\Sigma}_0 = \mathbf{A}\mathbf{A}^\top. \tag{20}$$

Hence, $\sigma_{g;g'}$ (covariance between y_g and $y_{g'}$) and $\sigma_{g;g} = \sigma_g^2$ (variance of y_g) are

$$\sigma_{g;g'} = \sum_{l=1}^{n} a_{g;l} a_{g';l} \ (g, g' = 1, \ldots, n). \tag{21}$$

For example, for $n = 2$ we get (remember that $\mathbf{A}$ is symmetric so $a_{g;g'} = a_{g';g}$)

$$\boldsymbol{\Sigma}_0 = \begin{bmatrix} a_{1;1}^2 + a_{1;2}^2 & a_{1;1}a_{2;1} + a_{1;2}a_{2;2} \\ a_{2;1}a_{1;1} + a_{2;1}a_{2;2} & a_{2;1}^2 + a_{2;2}^2 \end{bmatrix}. \tag{22}$$

Note that each element $a_{g;g'}$ $(= a_{g';g})$ affects the two variances and the covariance; we shall detail this characteristic in the next section. If we assume the Gaussian correlation function (2) and a single input x, then (19) becomes $R(x_i - x_{i'}; \theta^{(g)}) = \exp[-\theta^{(g)}(x_i - x_{i'})^2] = \exp[-\theta^{(g)} d_{i;i'}^2]$ with $d_{i;i'} = |x_i - x_{i'}|$ so we get

$$\mathrm{Cov}[\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})] = \mathbf{A} \begin{bmatrix} \mathbf{R}(d_{i;i'}; \theta^{(1)}) & 0 \\ 0 & \mathbf{R}(d_{i;i'}; \theta^{(2)}) \end{bmatrix} \mathbf{A}^\top$$

so $\mathrm{Cov}[\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})]$ is

$$\begin{bmatrix} a_{1;1}^2 \mathbf{R}(d_{i;i'}; \theta^{(1)}) + a_{1;2}^2 \mathbf{R}(d_{i;i'}; \theta^{(2)}) & a_{1;1}\mathbf{R}(d_{i;i'}; \theta^{(1)})a_{2;1} + a_{1;2}\mathbf{R}(d_{i;i'}; \theta^{(2)})a_{2;2} \\ a_{1;1}\mathbf{R}(d_{i;i'}; \theta^{(1)})a_{2;1} + a_{1;2}\mathbf{R}(d_{i;i'}; \theta^{(2)})a_{2;2} & a_{2;1}^2 \mathbf{R}(d_{i;i'}; \theta^{(1)}) + a_{2;2}^2 \mathbf{R}(d_{i;i'}; \theta^{(2)}) \end{bmatrix}. \tag{23}$$

Following Fricker et al. (2010) and Svenson & Santner (2010), we select $\mathbf{A}$ as the eigendecomposition of $\mathbf{\Sigma}_0$ while guaranteeing that $\mathbf{A}$ is PD. We therefore use the Cholesky transformation, $\mathbf{A} = \mathbf{L}\mathbf{L}^\top$. We should ensure that all the elements on the main diagonal of $\mathbf{L}$ are non-negative; i.e., we should impose the constraint $l_{i;i} \geq 0$ ($i = 1, ..., n$) in the MLE optimization.

Actually, Svenson & Santner (2010) apply *Restricted* MLE (RMLE) instead of MLE (for details on RMLE see Santner et al. (2003, pp. 66-67)). The RMLE $\widehat{\mathbf{A}}$ (which must be PD) and $\widehat{\mathbf{\Theta}}$ (the multivariate analogue of $\widehat{\boldsymbol{\theta}}$ defined below (6) in Step 1) minimize the following analogue of (6):

$$l(\mathbf{\Sigma_Y}, \boldsymbol{\mu}|\mathbf{Y}) = \ln|\mathbf{\Sigma_Y}| + \ln\left|\mathbf{F}^\top\mathbf{\Sigma_Y}^{-1}\mathbf{F}\right| + (\mathbf{Y} - \mathbf{F}\widehat{\boldsymbol{\mu}})^\top\mathbf{\Sigma_Y}^{-1}(\mathbf{Y} - \mathbf{F}\widehat{\boldsymbol{\mu}}). \quad (24)$$

In our next Monte Carlo experiments we shall use RMLE for univariate and multivariate Kriging for better comparison of the two methods. RMLE for univariate Kriging requires replacing m by $m - 1$ in (6), (8), and (9).

4 Monte Carlo laboratory: sampling from a GP

First we explain why we need a *laboratory* instead of *real* applications. Kriging is based on specific assumptions; e.g., Kriging assumes a GP. To analyze the performance of the resulting Kriging procedure, we should start with situations that satisfy these assumptions; a "laboratory" can fully satisfy all our assumptions. Real applications enable us to study the "robustness" of the Kriging method; i.e., how well does the method perform if not all its assumptions are completely satisfied? However, before we perform such robustness studies, we should examine the performance of Kriging in the case where all assumptions do hold. Moreover, real applications may be extremely expensive; i.e., a single simulation run may take hours or days, whereas in our lab a "simulation" run takes only (micro)seconds (depending on the computer hardware and software).

Kriging literature derives formulas for the estimated variance of the predictor in univariate and multivariate Kriging respectively. These formulas are popular, but we do not use them to compare univariate and multivariate Kriging because we estimate the MSE from the *known* I/O function for the simple systems that we simulate in our lab. Moreover, these formulas are biased because they ignore the variability caused by the estimation of the

parameters of the GP; see Den Hertog et al. (2006) and Kleijnen & Mehdad (2013).

To compare multivariate and univariate Kriging, we use the MSE criterion; this criterion gives the "optimal" Kriging predictor defined in (3), and is relevant in sensitivity analysis (not optimization) of simulation models. Furthermore, in our first example (Section 4.1) we briefly consider a second criterion; namely, the coverage of the 90% confidence interval for the predictor in univariate versus multivariate Kriging. Fricker et al. (2010) also use criteria closely related to our two criteria.

We wish to guarantee that the Kriging metamodel itself is a *valid* metamodel of the I/O function implied by the underlying simulation model. Therefore we generate the "simulation" observations $\mathbf{Y}$ from $\mathcal{N}_{mn}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\mathbf{Y}})$ defined in (11), to obtain the I/O data; these data are detailed in Table B.1 in Appendix B. To these I/O data we apply univariate and multivariate Kriging respectively, and compare their MSEs. Note that a similar Monte Carlo lab is used by Chen et al. (2012) for the "empirical evaluation" of their stochastic Kriging. In Section 4.1 we present a simple Monte Carlo example; in Section 4.2, we present a more complicated example.

4.1 Simple Monte Carlo example

After we specify that our lab consists of a GP, we must select values for the parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}_{\mathbf{Y}}$ in $\mathcal{N}_{mn}(\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\mathbf{Y}})$. In our simple example we specify a bivariate output so $n = 2$ and a single input so $k = 1$. To generate "simulation" data, we select $m = 10$ "old" I/O combinations; $m = 10$ agrees with the value $10k$ often recommended in the literature; also see the "practical guidelines" in Loeppky et al. (2009). Because space-filing designs are most popular in Kriging, we select these m values equi-spaced in the standardized experimental domain $0 \leq x_i \leq 1$. Consequently, we get $\mathbf{x}^\top = (0, 1/9, 2/9, \ldots, 8/9, 1)^\top$. We decide to predict the simulation outputs for m_0 "new" input values each halfway its two immediate neighbors, so we get $m_0 = 9$ and $\mathbf{x}_0^\top = (1/18, 3/18, \ldots, 17/18)^\top$. When we sample the GP, we should also sample the "true" simulation outputs at these new input values $\mathbf{x}_0^\top$; i.e., we sample $10 + 9 = 19$ bivariate outputs:

$$\boldsymbol{Y}_{(38\times 1)} = \begin{pmatrix} (y_{1;1}, y_{1;2})^\top \\ \vdots \\ (y_{19;1}, y_{19;2})^\top \end{pmatrix} \sim \mathcal{N}_{(38\times 1)}\left[\boldsymbol{\mu}_{(38\times 1)}\ , \boldsymbol{\Sigma}_{(38\times 38)}\right]. \qquad (25)$$

Furthermore, we select all $n = 2$ means equal to zero so in (25) we have $\boldsymbol{\mu}_{(38\times 1)} = (0, \ldots, 0)^\top$. We wish to experiment with "high" and "low" values for the variances σ_1^2 and σ_2^2. A problem, however, is that in the nonseparable dependence model, the variances—and cross-covariances and auto-covariances–depend on A in (19). In our experiments with two outputs, the values we select for the variances and the cross-covariance together with (22) imply

$$\boldsymbol{\Sigma}_0 = \begin{bmatrix} a_{1;1}^2 + a_{1;2}^2 & a_{1;1}a_{2;1} + a_{1;2}a_{2;2} \\ a_{2;1}a_{1;1} + a_{2;1}a_{2;2} & a_{2;1}^2 + a_{2;2}^2 \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \sigma_{1;2} \\ \sigma_{1;2} & \sigma_2^2 \end{bmatrix}, \qquad (26)$$

so the three variables $a_{1;1}$, $a_{1;2}$ $(= a_{2;1})$, and $a_{2;2}$ must satisfy three equations. We select $\sigma_1^2 = 1$, $\sigma_2^2 = 25$ (so $\sigma_2 = 5$, which quantifies variability better than its square, σ_2^2), and $\sigma_{1;2} = \rho^{(1;2)}\sigma_1\sigma_2 = 1$ so $\rho^{(1;2)} = 0.2$ and $\sigma_{1;2} = 4$ so $\rho^{(1;2)} = 0.8$; i.e., in our experiments we keep σ_1^2 and σ_2^2 fixed, while we experiment with a low and a high cross-correlation (the cross-correlation remains constant across input combinations). For the auto-covariances we assumed a Gaussian auto-correlation function and a single input, as derived in (23). In (23) we have already selected all elements of $\mathbf{A}$ (namely, $a_{1;1}$, $a_{1;2} = a_{2;1}$, and $a_{2;2}$) when selecting the variances and the cross-correlation. To further simplify our selection, we select equal Kriging parameters $\theta^{(1)} = \theta^{(2)} = \theta$ for the two outputs; this changes $\mathrm{Cov}[\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_{i'})]$ in (23) into

$$\begin{bmatrix} (a_{1;1}^2 + a_{1;2}^2)\mathbf{R}(d_{i;i'};\theta) & (a_{1;1}a_{2;1} + a_{1;2}a_{2;2})\mathbf{R}(d_{i;i'};\theta) \\ (a_{1;1}a_{2;1} + a_{1;2}a_{2;2})\mathbf{R}(d_{i;i'};\theta) & (a_{2;1}^2 + a_{2;2}^2)\mathbf{R}(d_{i;i'};\theta) \end{bmatrix}.$$

We wish to experiment with low and high auto-correlations. The Gaussian auto-correlation function (2) implies that $\mathrm{cor}(y_{i;1}, y_{i';1}) = \exp[-\theta(x_i - x_{i'})^2]$. Obviously, these correlations decrease with the distance $d_{i;i'} = |x_i - x_{i'}|$. These distances $d_{i;i'}$ vary with m (number of old equidistant input values in the experimental range) and the m_0 new input values to be predicted (which we selected halfway the old values). Given the input range $0 \leq x \leq 1$, these distances range between $1/[(m_0)/2] = 1/18$ (closest neighbors) and 1 (neighbors farthest apart). We decide to focus on the strongest auto-correlation between old input values $\rho^{(1)}(d_{\min}) = \exp[-\theta/9^2]$; e.g., $\rho^{(1)}(d_{\min}) = 0.2$ implies—after rounding—$\theta = 131$ and $\rho^{(1)}(d_{\min}) = 0.8$ implies $\theta = 18$, which are far away from the two extreme values 0 and ∞ discussed below (2). Altogether, Table 4.1 displays our four experiments combining "low" and "high" values for the cross-correlation $\rho^{(1;2)}$ and the maximum

Table 4.1: Design for first Monte Carlo example

	Experiments			
Correlation	1	2	3	4
cross: $\rho^{(1;2)}$	0.8	0.8	0.2	0.2
auto: $\rho^{(g)}(d_{\min})$	0.8	0.2	0.8	0.2

auto-correlation for output g denoted by $\rho^{(g)}(d_{\min})$ with g =1, 2. Note that Table 4.1 excludes zero or negative cross-correlations. We exclude zero correlation because discrete-event simulation with multiple outputs always gives correlated outputs as these outputs are driven by the same pseudorandom numbers. We exclude negative correlations, because the OR analysts usually know the signs of the correlations between the outputs of their simulation models; e.g., in queuing simulation, the average of the waiting time distribution and (say) the 90% quantile of that distribution are obviously positively correlated. If the analysts know that the correlation between two specific outputs is negative, then they simply take the negative values of one of these two outputs to make the correlation positive.

We decided to obtain $M = 100$ *macro-replicates*; i.e., we repeat our sampling—from the four different GPs in Table 4.1—100 times using non-overlapping pseudo-random number streams. We use these macro-replicates, to *verify* our computer code; i.e., we statistically test various intermediate results; namely, the means, variances, auto-correlations, and cross-correlations of the simulated bivariate GP output (25). This verification is detailed in Appendix D.

In practice, the simulation analysts have only "a single macro-replicate" to compute the RMLEs $\widehat{\boldsymbol{\mu}}$ and $\widehat{\boldsymbol{\Sigma}}$. In nonseparable multi-variate Kriging, these RMLEs are based on the likelihood function (24). This function may have many local maxima so the search for these RMLEs may get stuck on a local hill. To get initial estimates of $\widehat{\mathbf{A}}$ and $\widehat{\boldsymbol{\Theta}}$, Svenson (2011, pp. 314-315) uses a global optimizer; namely, the *genetic algorithm* (GA) in Forrester (2010); to get final estimates, he uses MATLAB's "fmincon". In our first Monte Carlo example, we use DACE for univariate Kriging; we combine DACE with the same GA and bounds that Svenson uses.

Given these RMLEs computed from the "old" I/O data, we can predict

the output for one of the new points (say) $x_{t'}$ with $t' = 1, \ldots, m_0$; i.e., in (3) we replace the unknown parameters by their RMLEs, which gives (say) $\widehat{\widehat{y}}_{t'}$. We compare this predicted value $\widehat{\widehat{y}}_{t'}$ with the "observed" value $y_{t'}$; the latter value is conditional on the m old observed values y_i because of (25), which uses the true parameters instead of their RMLEs. This gives the squared error (SE) at the new input value t' for output g in macro-replicate r where in this example $n = 2$, $m = 10$, $M = 100$:

$$\mathrm{SE}_{t';r}^{(g)} = (\widehat{\widehat{y_{t';r}^{(g)}}} - y_{t';r}^{(g)})^2 \text{ with } (g = 1, \ldots, n)\ (t' = 1, \ldots, m_0)\ (r = 1, \ldots, M). \tag{27}$$

Because we predict the output for m_0 new input combinations, we use this equation to compute the estimated *integrated MSE* (IMSE) for output g:

$$\widehat{\mathrm{IMSE}_r^{(g)}} = \frac{\sum_{t'=1}^{m_0} \mathrm{SE}_{t';r}^{(g)}}{m_0}.$$

Next we use this $\widehat{\mathrm{IMSE}_r^{(g)}}$ to compute its average over the M macro-replicates:

$$\overline{\widehat{\mathrm{IMSE}^{(g)}}} = \frac{\sum_{r=1}^{M} \widehat{\mathrm{IMSE}_r^{(g)}}}{M}. \tag{28}$$

Table 4.2 gives this average for univariate and multivariate Kriging, for the two outputs of our four experiments defined in Table 4.1. The standard errors —displayed in parentheses below the averages—show that the reported averages are quite accurate after $M = 100$ macro-replicates. All eight differences between these averages for univariate and multivariate Kriging—denoted by Difference in the table—are non-significant, where our null-hypothesis is that there is no difference between the MSEs of univariate and multivariate Kriging, for both responses and all four experiments. This significance we test through the t-statistic for differences estimated from correlated observations (caused by common random numbers used by the four experiments). We use an experimentwise type-I error rate $\alpha = 0.20$ combined with Bonferroni's inequality so the "per comparison" error rate is $(\alpha/2)/8 = 0.0125$. Furthermore, the point estimates show that univariate Kriging gives a smaller MSE for output 1 in all four experiments; for output 2 with its higher variance, univariate Kriging gives smaller MSE in two out of four experiments.

Comparisons of rows instead of columns in Table 4.2 shows that the averages are much bigger in rows 2 and 4. Table 4.1 shows that rows 2 and 4 have

a lower value for the auto-correlation; obviously, such a low auto-correlation implies that the Kriging predictor for a new point is less accurate when taking a weighted average of the old simulation outputs (the Kriging predictor is indeed a weighted average with weights determined by the distances between the new point and the old points). These differences between rows are so big that we do not need a statistical test to conclude that these differences are important. To explain the results in Table 4.2, we compute t-statistics from

Table 4.2: $\overline{\widehat{\text{IMSE}}^{(g)}}$ (with standard errors in parentheses) in univariate and multivariate Kriging, estimated from 100 macro-replicates

Experiment	Output 1 Univar.	Output 1 Multivar.	Output 1 Difference	Output 2 Univar.	Output 2 Multivar.	Output 2 Difference
1	0.000181	0.000238	-0.000057	0.004801	0.004784	0.000017
	(0.000027)	(0.000055)	(0.000039)	(0.000685)	(0.000589)	(0.000475)
2	0.239103	0.239563	-0.000460	5.762738	5.813574	-0.050837
	(0.013688)	(0.014082)	(0.002252)	(0.336574)	(0.338779)	(0.042030)
3	0.000179	0.000198	-0.000018	0.004187	0.004012	0.000175
	(0.000027)	(0.000037)	(0.000013)	(0.000665)	(0.000593)	(0.000133)
4	0.239072	0.240658	-0.001585	5.712830	5.732354	-0.019523
	(0.013691)	(0.013929)	(0.002042)	(0.340724)	(0.339981)	(0.032913)

the $M = 100$ RMLEs for the Kriging parameters $(\mu_g, \sigma_g^2, \theta^{(g)}, \sigma_{g;g'})$ where $\sigma_{g;g'} = 0$ in univariate Kriging. Our null-hypothesis H_0 states that the expected value of a RMLE equals the true value of the corresponding Kriging parameter; e.g., $H_0 : \widehat{\mu_g} = \mu_g$. Table 4.3 shows whether such a t-test rejects H_0; i.e., the superscript $*$ denotes that the t-statistic is significant at the 5% significance level. This table implies that for univariate Kriging we should reject H_0 only for $\theta^{(g)}$, whereas for multivariate Kriging we should reject H_0 for $\theta^{(g)}$, σ_g^2, and $\sigma_{g;g'}$. So multivariate Kriging gives inaccurate estimates of the Kriging parameters. These inaccurate estimates may result from the search in a space with more dimensions when solving a nonconvex problem. Svenson & Santner (2010) also mention that RMLE in multivariate Kriging requires a search in higher dimensions than univariate Kriging does, because the latter assumes zero cross-correlations; so the former search might actually result in poor estimates of the Kriging parameters. Svenson & Santner (2010) and Fricker

et al. (2010) give numerical results for several examples suggesting that multivariate Kriging may not improve MSE dramatically relative to application of univariate Kriging to multiple outputs. To further investigate these numeri-

Table 4.3: t-tests for RMLE in univariate and multivariate Kriging

			Univariate Kriging				
Experiment	μ_1	μ_2	θ_1	θ_2	σ_1^2	σ_2^2	σ_{12}
1	1.1510	1.1529	2.0268*	2.4019*	0.5672	-0.8968	-
2	1.4495	1.4557	-6.3886*	-4.9355*	-0.3692	-0.1543	-
3	1.1501	0.4909	1.9842	2.9331*	0.5652	-2.5660*	-
4	1.4502	0.7710	-6.3868*	-4.1674*	-0.3822	-1.5303	-
			Multivariate Kriging				
Experiment	μ_1	μ_2	θ_1	θ_2	σ_1^2	σ_2^2	σ_{12}
1	1.2284	1.2597	1.9355	-0.2059	2.4377*	2.2318*	2.1643*
2	1.4572	1.4027	-8.3775*	-8.6159*	3.1470*	2.8997*	2.9758*
3	0.8132	0.4204	0.3007	1.1068	1.2741	0.6756	1.5367
4	1.5185	0.8312	-9.1144*	-6.6839*	3.4481*	1.5533	1.9560

cal results, we run univariate and multivariate Kriging with the *true* Kriging parameters—which is easy in Monte Carlo experiments and impossible in real experiments. We point out that univariate Kriging uses 100% accurate information on $\widehat{\sigma_g^2}$ and $\widehat{\theta_j^{(g)}}$, but that information is incomplete because it ignores the cross-correlations $\widehat{\sigma_{g;g'}}$ and consequently univariate Kriging uses the wrong $\widehat{\mu}_g$. The "old" and "new" outputs vary over the M macro-replicates, because they are sampled from (25). This gives Table 4.4, which displays the average performance defined in (28); this performance is the same for univariate and multivariate Kriging. Comparing this table with Table 4.2 shows that the performance is better when using the true Kriging parameters instead of their RMLEs, as we expected. However, IMSEs in Table 4.4 are not significantly better than IMSEs in Table 4.2. To explain that in this example univariate and multivariate Kriging give the same performance when they use the true Kriging parameters, we study the only difference between univariate and multivariate Kriging; namely, $\mathbf{\Sigma}_{0;m;n}\mathbf{\Sigma}_{\mathbf{Y}}^{-1}$ in (15). Univariate Kriging assumes zero cross-covariances. We find that the corresponding elements in multivariate Kriging—using the true Kriging parameters—are

Table 4.4: $\overline{\widehat{\text{IMSE}^{(g)}}}$ in univariate and multivariate Kriging with true GP parameters

Experiment	Output 1	Output 2
1	0.000123	0.003318
2	0.235071	5.756298
3	0.000123	0.00299
4	0.235071	5.708968

virtually zero. For example, when predicting the output for the first element of $\mathbf{x}_0^\top = (1/18, 3/18, \ldots, 17/18)^\top$ in macro-replicate 1 of experiment 1 (with $\rho^{(1:2)} = 0.8$), multivariate Kriging gives values between 10^{-16} and 10^{-13}. We also compute these values for $\rho^{(1:2)}= 0.95$ and again find virtually zero values; detailed results are given in Table E.1 of Appendix E. These results are typical for this example; i.e., in the more realistic example of Section 4.2 we shall find different results.

Our conclusion is that univariate Kriging is simpler than multivariate Kriging, and that multivariate Kriging does not perform better than univariate Kriging, even when multivariate Kriging would know the true GP parameters.

Besides the MSE, Fricker et al. (2010) study the *coverage* of the confidence interval of the Kriging predictor. A popular 90% two-sided confidence intervals for $\widehat{\widehat{y_{t';r}^{(g)}}}$ is

$$\widehat{\widehat{y_{t';r}^{(g)}}} \pm 1.64\sqrt{\text{MSPE}_{t';r}^{(g)}}$$

where 1.64 is the 0.95 quantile of the standard Gaussian density and $\text{MSPE}_{t';r}^{(g)}$ follows from (10) and (17). Obviously, for given t' and g values, macro-replicate r gives an interval that does or does not cover the true value $y_{t';r}^{(g)}$; from the M macro-replicates we compute the estimated coverage. The coverage of this 90% confidence interval turns out to be too low, for any t and g (box plots are available from the authors). This low coverage may be caused by the classic variance of the Kriging predictor, which ignores consequences of estimating the Kriging parameters; see again Den Hertog et al. (2006) and Kleijnen & Mehdad (2013).

Finally, Fricker et al. (2010) suggest that the relative performance of multivariate Kriging may improve when "the" output is a *function* of the individual cross-correlated outputs. Therefore we also experiment with the *sum* and the *product*, $y^{(3)} = y^{(1)} + y^{(2)}$ and $y^{(4)} = y^{(1)}y^{(2)}$. Appendix F (Table F.1) suggests that univariate Kriging gives smaller MSE for all $n = 4$ outputs and all four experiments in Table 4.1 except for $y^{(4)}$ in experiment 1. Univariate Kriging gives better coverage than multivariate Kriging, but still below the nominal value. Note that we do know the true values of the Kriging parameters for $y^{(1)}$ and $y^{(2)}$, but not for $y^{(3)}$ and $y^{(4)}$. Furthermore, there are more Kriging parameters to be estimated; namely, $\sigma_{g,g'}$ $(g, g' = 1, \ldots, n = 4)$ and $\theta^{(g)}$. The appendix (Table F.2) suggests that multivariate Kriging has more significant differences between the estimated and the true parameter values.

4.2 More complicated Monte Carlo example

In this subsection we summarize our second type of Monte Carlo example; namely, an example with $d = 5$ inputs and $n = 3$ outputs that is inspired by the *simple climate model* (SCM) case study in Fricker et al. (2010). For this example we select $m = 57$ old I/O data $(\mathbf{X}, \mathbf{W})$ and $m_0 = 93$ new data $(\mathbf{X}_0, \mathbf{W}_0)$; these data we received from one of the coauthors (namely, Urban). For our Kriging computations we use Svenson's code. So, from the old data $(\mathbf{X}, \mathbf{W})$ we compute $\widehat{\boldsymbol{\psi}}$, the RMLE of the GP parameters. This $\widehat{\boldsymbol{\psi}}$ has 24 elements; namely, the three means $\widehat{\mu_g}$ $(g = 1, 2, 3)$, the three variances $\widehat{\sigma_g^2}$, the $5 \times 3 = 15$ auto-correlations $\widehat{\theta_j^{(g)}}$ $(j = 1, ..., d = 5)$, and the three cross-covariances $\widehat{\sigma_{g;g'}}$.

We observe that the three outputs have indeed positive cross-correlations; i.e., from the simulation outputs $\mathbf{W}$ we compute the classic estimates which do not assume a GP:

$$r^{(g;g')} = \frac{\sum_{t=1}^{m}(w_t^{(g)} - \overline{w^{(g)}})(w_t^{(g')} - \overline{w^{(g')}})}{m}. \tag{29}$$

This gives $r^{(1;2)} = 0.47$, $r^{(1;3)} = 0.55$, and $r^{(2;3)} = 0.80$; these estimates are scale-free. The estimates (29) should be distinguished from $\widehat{\rho^{(g;g')}}$, which denote the RMLE computed for the multivariate GP with scaled old data. These computations give $\widehat{\rho^{(1;2)}} = 0.50$, $\widehat{\rho^{(1;3)}} = 0.53$, and $\widehat{\rho^{(2;3)}} = 0.82$, which agree very well with the classic estimates.

Because we scale the I/O data, the simulation outputs have zero means. The $n = 3$ estimated GP *means* $\widehat{\mu}_g$ turn out to be virtually zero, in both multivariate and univariate Kriging. For the computations of univariate Kriging we apply Svenson's code *per output*, ignoring cross-correlations; i.e., assuming these correlations are zero. The estimated variances in multivariate Kriging $\widehat{\sigma}_g^2$ are 5.98, 3.48, and 3.62; i.e., output 1 has a bigger variance. We point out that univariate Kriging gives different estimates; namely, 6.38, 4.29, and 7.09; we emphasize that univariate Kriging assumes zero cross-variances so it is to be expected that its variance estimates are different. The 15 estimated auto-correlation coefficients $\widehat{\theta_j^{(g)}}$ differ in multivariate Kriging and univariate Kriging.

In our second Monte Carlo example, the true GP is the GP with the parameters that we estimated for the SCM case study using multivariate Kriging; e.g., $\widehat{\rho^{(1;2)}} = 0.50$. Analogous to (25) we sample $\mathbf{Y}$ from the multivariate normal distribution; this $\mathbf{Y}$ has $(57 + 93) \times 3 = 450$ elements. This gives the IMSE defined analogously to (28). We use $M = 30$ macro-replicates, which require approximately 15 hours of computer time on our PC. Comparing columns 2 and 4 of Table 4.5 shows that univariate Kriging gives a much smaller MSE than multivariate Kriging. Like we do in our first Monte

Table 4.5: $\overline{\widehat{\text{IMSE}^{(g)}}}$ (with standard errors in parentheses) in univariate and multivariate Kriging, estimated from 30 macro-replicates

Output	Multivar. with estimated GP par.	Multivar. with true GP par.	Univar. with estimated GP par.	Univar. with true GP par.
1	33,953	1,487	922	1,699
	(4,685)	(256)	(73)	(215)
2	16,392	700	455	1,529
	(2,202)	(127)	(37)	(263)
3	2.090	0.126	0.060	0.846
	(0.266)	(0.026)	(0.004)	(0.103)

Carlo experiment, we also compute the IMSE assuming true GP parameters $\widehat{\sigma}_g^2$, $\widehat{\theta_j^{(g)}}$, and $\widehat{\sigma_{g;g'}}$; notice that $\widehat{\mu}_g$ is computed through (16). Comparing columns 2 and 3 shows that using these true parameters drastically decreases the IMSE of multivariate Kriging. Comparing columns 3 and 4 shows that

multivariate Kriging with true GP parameters does not give a smaller IMSE than univariate Kriging with estimated parameters. Comparison of these two columns shows that multivariate Kriging's inferior performance is not due to the estimation of more parameters. We conjecture that multivariate Kriging suffers from an inherent bad property—like separable models, which have the Markov property, which implies that the cross-covariance between the outputs does not help. Comparing columns 3 and 5 shows that using the true parameters gives univariate Kriging with higher IMSE than multivariate Kriging; in Section 4.1 we have already pointed out that univariate Kriging uses 100% accurate but incomplete information on the GP parameters. We point out that the first example used assumptions so simplistic that univariate Kriging was not affected by the incompleteness of the information on the RMLE of the GP parameters.

5 Conclusions and future research

In this article we compare univariate and multivariate Kriging metamodels for simulation models with multiple outputs. A major problem of multivariate Kriging is ensuring that the (symmetric) covariance matrix of all the observed simulation outputs remains positive-definite; to solve this problem, we apply a nonseparable dependence model that was originally proposed by Fricker et al. (2010). We compare the resulting multivariate Kriging with univariate Kriging per type of simulation output; univariate Kriging ignores the cross-correlations between the multiple simulation outputs. To compare these two Kriging types, we perform some Monte Carlo experiments in which we guarantee that all the assumptions of multivariate Kriging are satisfied. We use these experiments to estimate the MSEs of multivariate and univariate Kriging. These experimental results suggest that the simpler univariate Kriging gives lower MSE than the more complicated multivariate Kriging; one explanation is that multivariate Kriging requires the estimation of additional Kriging parameters—namely, the cross-correlations between the simulation outputs—which affects the estimates of all parameters. To check this explanation, we run additional Monte Carlo experiments replacing the estimated Kriging parameters by their true values, which are known in a Monte Carlo experiment. We then find that multivariate Kriging with the true GP parameters still does not perform better than univariate Kriging. We conjecture that nonseparable models created through LMC have some

inherent property that causes inferior performance. In example 1 we briefly examine the coverage, and conclude that both multivariate and univariate Kriging give coverages lower than the nominal (90%) value; multivariate Kriging does not improve this coverage.

Future research may address the following topics.

- Multivariate Kriging may serve as a metamodel not only for simulation models with multiple outputs but also for multi-fidelity simulation, discussed in Section 1 (type 2). Our Monte Carlo lab can be easily adapted to model such types of simulation. Besides the output of interest, the modelers may also estimate its gradient; also see Section 1 (type 3).

- The goal of simulation may be optimization instead of sensitivity analysis. This optimization might replace MSE by a criterion such as used in efficient global optimization (EGO), but adapted for multivariate optimization. This optimization may use constrained optimization, selecting one output as the goal variable and satisfying constraints on the $(n-1)$ remaining outputs. An alternative for this constrained optimization is multi-objective Pareto optimization.

- In our Monte Carlo experiments we guaranteed that Kriging gives a valid metamodel, but in practice we may improve the validity of the metamodel by replacing ordinary Kriging by "universal" Kriging which uses a linear regression model instead of the constant μ in (1); see Fricker et al. (2010).

- In our Monte Carlo experiments we may replace the Gaussian correlation function by some other correlation function to generalize our results; in practice, such a function may improve the validity of the Kriging metamodel.

- If indeed multivariate Kriging does not outperform univariate Kriging in deterministic simulation, then it does not seem interesting to extend multivariate Kriging from deterministic simulation to more complicated (random) discrete-event simulation.

Acknowledgments

Joshua Svenson and Tom Santner (Ohio State) shared their Kriging code with us. Nathan Urban shared his data on the climate model that we used in our second Monte Carlo experiments. David Ginsbourger referred us to Álvarez et al. (2011). Two anonymous reviewers gave detailed comments that helped us to improve the original version of our paper.

A List of major symbols

A list of major symbols is given in Table A.1, in alphabetical order with Latin symbols before Greek symbols; bold letters denote matrices and vectors.

B I/O data for multivariate Kriging

Table B.1 shows the I/O data for multivariate Kriging.

C Simplest example: two outputs and one input

The simplest example of a simulation model with multiple types of outputs is a model with *two* types only; say, y_1 and y_2. Furthermore, the simplest model has a *single* input (say) x. Figure C.1 illustrates this example, assuming it is simulated for $m = 3$ input values. Two input values are relatively close together; namely, x_1 and x_2. If we consider the bivariate output (y_1, y_2) at a given input value such as x_1, then we see that these two outputs $y_1(x_1)$ and $y_2(x_1)$ are cross-correlated; the figure shows this correlation through vertical dotted curves. Moreover, Kriging implies that a given type of output such as y_1 is correlated with itself when observed at different input combinations; e.g., $y_1(x_1)$ and $y_1(x_2)$ are correlated. This correlation is called auto-correlation. The figure shows this auto-correlation through the (tilted horizontal) solid lines. The other type of output y_2 is also auto-correlated, but we do not show this correlation in the figure, to keep the figure simple. Obviously, outputs such as $y_1(x_3)$ and $y_2(x_1)$ are also correlated: in the figure we can follow the line from $y_1(x_3)$ and $y_1(x_1)$, and then the vertical curve to $y_2(x_1)$.

Table A.1: List of major symbols

Symbol	Meaning
$c_{0,i}$	covariance between outputs of old input combination i and new combination 0
i	index with range $1, \ldots, m$
g	index with range $1, \ldots, n$
k	number of simulation inputs
m	number of "old" simulated input combinations
m_0	number of "new" simulated input combinations
n	number of output types per input combination
$r^{(g;g')}$	estimated correlation coefficient for outputs g and g'
$\boldsymbol{R}$	correlation matrix
t	index with range $1, \ldots, 2m-1$
t'	index with range $1, \ldots, m_0$
$\mathbf{W}$	simulation outputs at old input combinations
$\mathbf{W_0}$	simulation outputs at new input combinations
$\mathbf{x}$	input combination
$\mathbf{x_0}$	new input combination
y	univariate output of a simulated input combination
$\boldsymbol{Y}$	multivariate GP output of a simulated input combination
$y_{i;g}$	output of type g for input combination i
$\widehat{y_0}$	univariate Kriging predictor of output of new input combination x_0
z	stationary Gaussian process with zero mean
$\theta_j^{(g)}$	importance of input j for the auto-correlation in outputs g
μ	mean univariate output
μ_g	mean of output type g
$\rho_{i,i'}^{(g;g')}$	correlation between outputs g and g' at input combinations i and i'
$\sigma_{i,i'}^{(g;g')}$	covariance between outputs g and g' at input combinations i and i'
σ_z^2	variance of univariate z
$\boldsymbol{\Sigma}$	covariance matrix of univariate output
$\mathbf{\Sigma_0}$	covariance of $\mathbf{Y}$
$\mathbf{\Sigma_{0;m;n}}$	$n \times nm$ covariance between $\mathbf{Y}(\mathbf{x}_0)$ and $\boldsymbol{Y}$
$\mathbf{\Sigma_Y}$	covariance matrix of multivariate output $\boldsymbol{Y}$
$\boldsymbol{\psi}$	GP parameters

Table B.1: I/O data for multivariate Kriging

Combination	Input	Output
1	$x_{1;1}, \ldots, x_{1;k}$	$y_{1;1}, \ldots, y_{1;n}$
2	$x_{2;1}, \ldots, x_{2;k}$	$y_{2;1}, \ldots, y_{2;n}$
$\vdots$	$\vdots$	$\vdots$
i	$x_{i;1}, \ldots, x_{i;k}$	$y_{i;1}, \ldots, y_{i;n}$
$\vdots$	$\vdots$	$\vdots$
m	$x_{m;1}, \ldots, x_{m;k}$	$y_{m;1}, \ldots, y_{m;n}$

Mathematically, in our example with $k = 1$ the Gaussian correlation function for output of type 1 becomes

$$\text{cor}[y_1(x_i), y_1(x_{i'})] = \exp[-\theta^{(1)}(x_i - x_{i'})^2] \text{ with } \theta^{(1)} \geq 0. \tag{30}$$

In the example the input x has values such that the outputs of type 1 $y_1(x_1)$ and $y_1(x_2)$ have higher positive auto-correlation than $y_1(x_1)$ and $y_1(x_3)$ have. Notice that if the input has the same value $x_i = x_{i'} = x$ in (30), then

$$\text{cor}[y_1(x), y_1(x)] = \exp[-\theta^{(1)}(x-x)^2] = \exp(-\theta^{(1)} \times 0) = \exp(-0) = 1/1 = 1, \tag{31}$$

whatever the value of $\theta^{(1)}$ is. So, these outputs have the highest positive (auto)correlation—which makes perfect sense in deterministic simulation. For output of type 2 we replace $\theta^{(1)}$ by $\theta^{(2)}$ in (30).

Note that stationarity of the process implies that only the distance in the input space matters, not the direction; e.g., $y_{1;1}$ and $y_{2;2}$ are as strongly correlated as $y_{2;1}$ and $y_{1;2}$ are

D Verification of first Monte Carlo example

In the first Monte Carlo example, we have $n = 2$ outputs $y_{t;g}$ for $m = 10$ old input combinations and $m_0 = m - 1 = 9$ new combinations halfway the two neighboring old combinations, so $t = 1, \ldots, 2m - 1$ and $g = 1, \ldots, n$. Univariate and multivariate Kriging *standardize* the old outputs y_i through

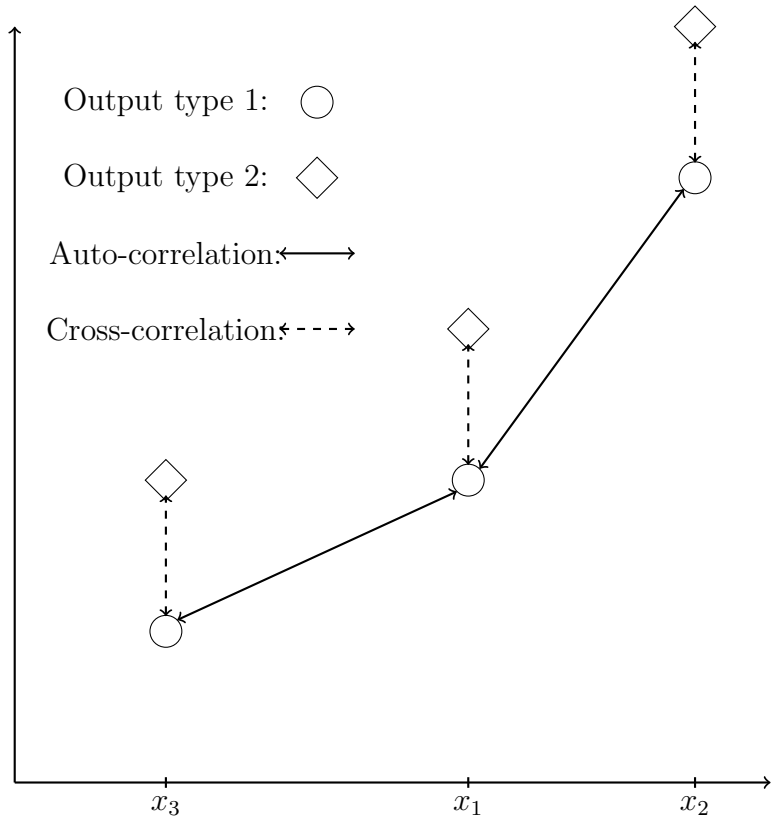

Figure C.1: Simplest example: bivariate output and single input

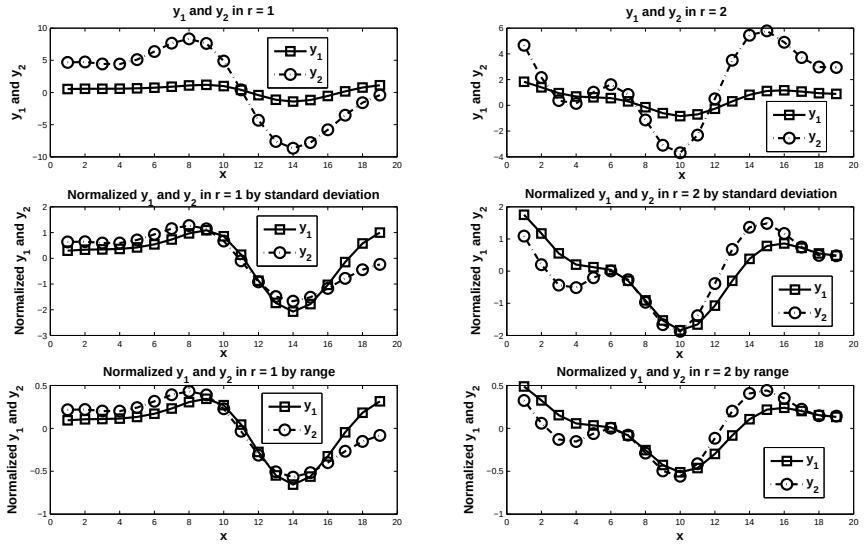

Figure D.1: $y_{t;g}$ for macro-replicates 1 and 2 in experiment 1

the linear transformation $(y_i - \overline{y})/s$ where $\overline{y}$ and s are shorthand notations for

$$\overline{y_g}(m) = \frac{\sum_{i=1}^{m} y_{i;g}}{m} \text{ with } g = 1, \ldots, n \tag{32}$$

and

$$s_g(m) = \sqrt{\frac{\sum_{i=1}^{m} [y_{i;g} - \overline{y_g}(m)]^2}{m}}. \tag{33}$$

Notice that the correlation coefficients are not affected by the linear transformations implied by standardization; however, the Kriging parameters (such as $\theta^{(g)}$) are affected, so we shall present empirical results for non-standardized outputs for both univariate and multivariate Kriging.

For our simple example, the Monte Carlo experiments give Figure D.1; this figure displays $y_{t;g}$ with $t = 1, \ldots, 19$ and $g = 1, 2$ for macro-replicates 1 and 2 of experiment 1. These plots suggest that the non-standardized outputs have indeed zero means, and that output 2 has higher variability. The three plots also suggest that the outputs are auto-correlated (plots for higher auto-correlation are not displayed). We augment this visual analysis as follows. We derive the following four statistical tests to verify that our Monte Carlo laboratory has no errors; i.e., our null-hypothesis is that there

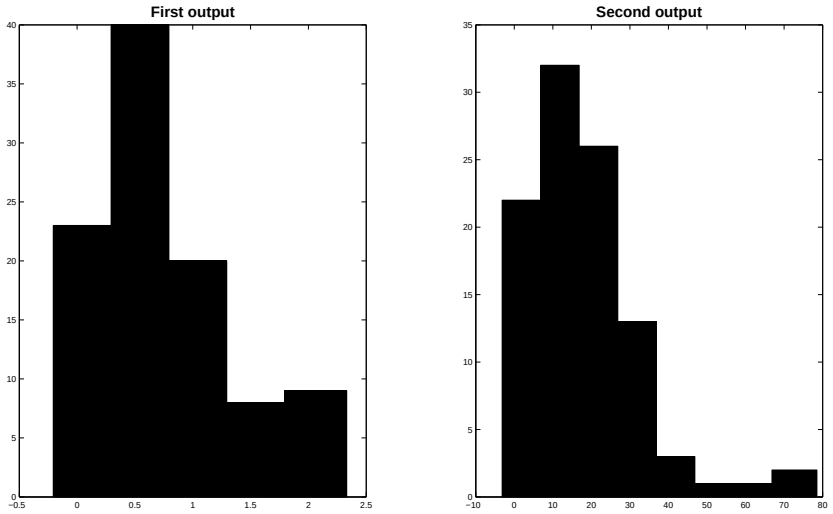

Figure D.2: $s_1^2(19)$ and $s_2^2(19)$ in 100 macro-replicates of experiment 1

are no such errors.

(i) We test whether the *averages* $\overline{y_g}(m)$ defined in (32) are indeed close to the true value $\mu_g = 0$. Unfortunately, $s_g^2(m)$ defined in (33) is not an unbiased estimator of the variance σ_g^2, because the outputs at different input combinations are positively correlated; i.e., $s_g^2(m)$ underestimates, as is illustrated by Figure D.2 which is a histogram of all M estimates.

Therefore we test our computer code using the M macro-replicates, which by definition are independently identically distributed (IID); i.e., defining $y_{t;g;r}$ as output g at input combination t in macro-replicate r gives both

$$\overline{y_{t;g}}(M) = \frac{\sum_{r=1}^{M} y_{t;g;r}}{M} \ (t = 1, \ldots, 2m-1; g = 1, \ldots, n) \tag{34}$$

and the unbiased variance estimators for output g of input combination t

$$s_{t;g}^2(M) = \frac{\sum_{r=1}^{M} [y_{t;g;r} - \overline{y_{t;g}}(M)]^2}{M-1}. \tag{35}$$

To test the null-hypothesis that the mean output g is zero, we use the Student t-statistic with $M-1$ degrees of freedom (DF):

$$t_{M-1}^{(t;g)} = \frac{\overline{y_{t;g}}(M) - 0}{s_{t;g}(M)/\sqrt{M}}. \tag{36}$$

Because $M = 100$, we use the standard Gaussian distribution $N(0,1)$ for the $t_{M-1}^{(t;g)}$ distribution. In experiment 1 of the four experiments in Table 1 of thee main text we find that for input value 1 and output 1, (36) gives $t_{100-1}^{(1;1)} = 0.123085/0.1162399 = 1.06$. Altogether we have $(2\times19 =)$ 38 t-values, so we use Bonferroni's inequality to obtain an experimentwise type-I error rate that does not exceed α; i.e., we replace $\alpha/2$ (two-sided test) by $\alpha/(2\times38)$ and we select $\alpha = 0.20$; this "experimentwise" error rate is higher than the classic 10% or 5%. We find that none of the 38 observed t-values is significant at any reasonable type-I error probability (the expected number of rejections for $\alpha = 0.20$ is $38 \times 0.20/(2 \times 38) = 0.1$, less than one).

(ii) We test whether the *variance* of output g is indeed σ_g^2. The GP assumption implies that the variances remain constant at all $2m-1$ input combinations t. Nevertheless, we should not "pool" the $2m-1$ variance estimators, as they are not independent because $y_{t;g;r}$ and $y_{t';g;r}$ are correlated. We use χ^2_{M-1}, which denotes the chi-square statistic with $(M-1)$ DF. Again using Bonferroni's inequality, we replace $\alpha/2$ by $\alpha/(2 \times 19)$ in the example, and select $\alpha = 0.20$; so $\alpha/38 = 0.005$. In our example, only one of the 19 points in experiment 4 gives a significant result; we decide not to reject our Monte Carlo experiments.

(iii) Analogously to the variance estimator (35) we define the *covariance* estimator

$$s_{t;t'}^{(g;g')}(M) = \frac{\sum_{r=1}^{M}[y_{t;g;r} - \overline{y_{t;g}}(M)][y_{t';g';r} - \overline{y_{t';g'}}(M)]}{M-1}, \tag{37}$$

which defines estimators for auto-covariances ($g = g'$) and cross-covariances ($g \neq g'$); obviously this equation is a special case of (37). Using (37), we obtain the (scale-free) estimated *linear correlation coefficients*

$$\widehat{\rho}_{t;t'}^{(g;g')}(M) = \frac{s_{t;t'}^{(g;g')}(M)}{s_{t;g}(M)s_{t';g'}(M)}. \tag{38}$$

This equation implies that the cross-correlation coefficient between outputs g and g' at combination t—estimated from the M macro-replicates—is

$$\widehat{\rho}_{t}^{(g;g')}(M) = \frac{s_{t}^{(g;g')}(M)}{s_{t;g}(M)s_{t;g'}(M)}. \tag{39}$$

We wish to test whether this coefficient deviates significantly from its expected true value ρ,which is determined by σ_1^2, σ_2^2, and $\sigma_{1;2}$. For this test we

use Press et al. (1992, pp. 637-638) to define

$$\frac{z-\omega}{\sigma_z} \text{ with } z=\frac{1}{2}\ln\frac{1+\widehat{\rho}}{1-\widehat{\rho}},\ \omega=\frac{1}{2}[\ln\{\frac{1+\rho}{1-\rho}\}+\frac{\rho}{M-1}],\ \sigma_z=\frac{1}{\sqrt{M-3}}. \tag{40}$$

This $(z-\omega)/\sigma_z$ has a $N(0,1)$ distribution asymptotically. We again use Bonferroni's inequality. In our example, we find that none of the estimated cross-correlations differs significantly from the known values $\rho^{(1;2)}$ in the four experiments.

Because (39) is a ratio estimator, we know that this estimator is *biased*. We also know that *jackknifing* reduces the bias of such an estimator; see the overview including references in Kleijnen (2008, pp. 81-84). Jackknifing is a simple statistical technique, which in this case works as follows. First we compute the so-called pseudovalue, which is a weighted combination of the original estimator $\widehat{\rho}_t^{(g;g')}(M)$—abbreviated to $\widehat{\rho}_t^{(g;g')}$—and the estimator deleting macro-replicate r denoted by $\widehat{\rho}_{t;-r}^{(g;g')}(M-1)$ or briefly $\widehat{\rho}_{t;-r}^{(g;g')}$:

$$J_r=M\widehat{\rho}_t^{(g;g')}-(M-1)\widehat{\rho}_{t;-r}^{(g;g')}\ (r=1,\ldots,M). \tag{41}$$

Next we compute the average pseudovalue $\overline{J}=\sum J_r/M$, which is expected to have less bias. We use $s^2(\overline{J})=s^2(J)/M$ where $s^2(\overline{J})$ is the estimated variance of this average pseudovalue:

$$s^2(\overline{J})=\frac{\sum_{r=1}^{M}(J_r-\overline{J})^2}{(M-1)M}.$$

Finally, we compute the following $(1-\alpha)$ two-sided confidence interval for $\rho_t^{(g;g')}$:

$$\overline{J}\pm t_{M-1;1-\alpha/2}s(\overline{J}). \tag{42}$$

We obtain (mutually correlated) estimates $\widehat{\rho}_t^{(1;2)}$ at the various input values t in each of our experiments, so when we use (40) or (42) we again apply Bonferroni's inequality—analogously to (i) and (ii). In all our four experiments we may accept the results.

(iv) Finally, we examine $\rho_{t;t'}^{(g)}(M)$, which denotes the *auto-correlation* between outputs g at locations t and t' estimated from M macro-replicates; see (38). Actually, we have m old input values and $m-1$ new values so altogether we have $2m-1$ outputs y_t $(t=1,\ldots,2m-1)$. Consequently, we have $2m-2$ observations $(y_t^{(g)},y_{t+1}^{(g)})$ with the minimum distance (say) $h_1=$

$|x_t - x_{t+1}| = 1/(2m-1)$; as the distance h between the input values of the outputs increases, the number of observations decreases. This enables the following estimators of the covariances with distance h for output g:

$$\widehat{c_h^{(g)}} = \frac{\sum_{t=1}^{2m-h-1}[y_{t;g} - \overline{y_{t;g}}][y_{t+h;g} - \overline{y_{t+h;g}}]}{2m-1} \; (h = 0, 1, \ldots, 2m-2) \qquad (43)$$

where—because of the GP assumption—we use $\overline{y_{t;g}} = \overline{y_{t+h;g}} = \overline{y_g}$; we use the denominator $(2m-1)$ because MATLAB follows Box et al. (1994), who claim that this denominator minimizes the MSE (not the bias) of the covariance estimators. So the auto-correlations are

$$\widehat{\rho}_h^{(g)} = \frac{\widehat{c_h^{(g)}}}{\widehat{c_0^{(g)}}}. \qquad (44)$$

This equation gives Figure D.3, which displays $\widehat{\rho}_{h;r}^{(g)}$ where r refers to macro-replicate r. This figure demonstrates that the low θ of experiments 1 and 3 does give a high estimated auto-correlation $\widehat{\rho}_h^{(g)}$ for small distances h (h = 0, 1, 2); for bigger distances, however, some auto-correlations become negative. Note that high correlation implies that the sample gives relatively little information. Next we fit a curve to these $\widehat{\rho}_{h;r}^{(g)}$ per macro-replicate.

We apply nonlinear regression analysis using MATLAB, which follows Seber & Wild (2003). MATLAB fits a curve $\alpha_1 + \alpha_2 e^{-\theta_R h^2}$, which generalizes the Gaussian correlation function through α_1 and α_2 so we expect $\widehat{\alpha_1} = 0$ and $\widehat{\alpha_2} = 1$. This results in the nonlinear regression estimate $\widehat{\theta_{r;R}^{(g)}}$, which should be distinguished from the RMLE $\widehat{\theta_r^{(g)}}$. Our example gives the histogram with $M = 100$ values in Figure D.4. The null-hypothesis $H_0 : \widehat{\theta_{r;R}^{(g)}} = \theta^{(g)}$ is rejected by both the t-test and the sign-test, but the actual difference $\widehat{\theta_{r;R}^{(g)}} - \theta^{(g)}$ is not really big. Similar conclusions hold for α_1 and α_2. The $\widehat{\theta_{r;R}^{(g)}}$ does change much when the true value $\theta^{(g)}$ changes from 18 to 131.

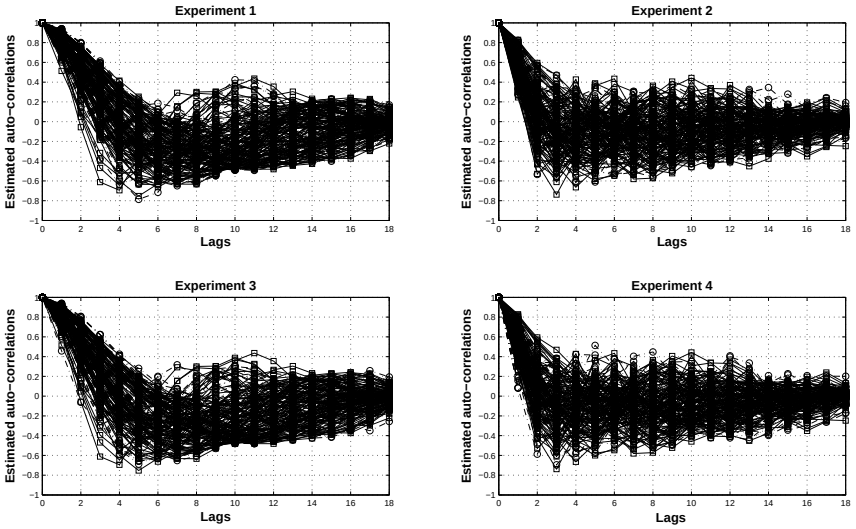

Figure D.3: Estimated auto-correlations versus lags in 100 macro-replicates

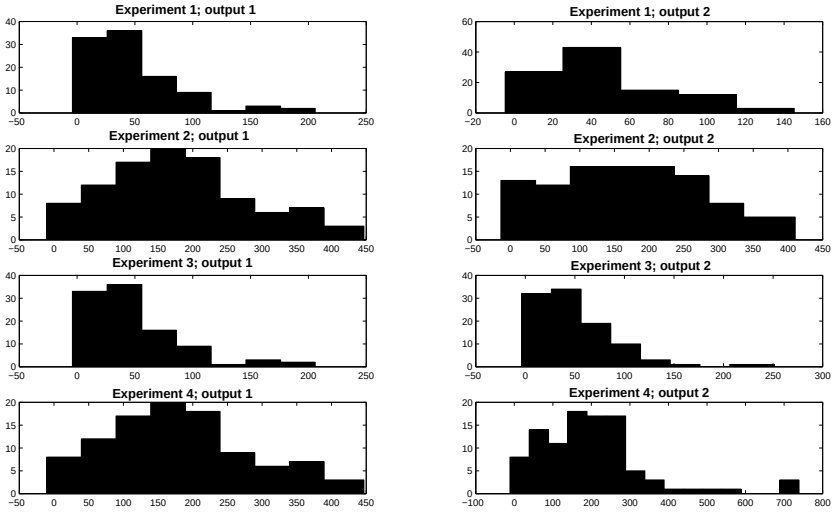

Figure D.4: $\widehat{\theta^{(g)}_{r;R}}$ for all the four experiments; $g = 1$ and 2

We conclude that (i) the estimated auto-correlations for different dis-

tances are biased because they are ratio estimators and they use a denominator that minimizes MSE instead of bias; (ii) these estimators are poor in case of high correlation.

E $\boldsymbol{\Sigma_{0;m;n}\Sigma_{Y}^{-1}}$ in univariate and multivariate Kriging

Table E.1 gives $\boldsymbol{\Sigma}_{0;m;n}\boldsymbol{\Sigma}_{\mathbf{Y}}^{-1}$ when predicting the output for $x_0 = 1/18$ in macro-replicate 1 of experiment 1, for cross-correlations 0.80 and 0.95; to improve the layout we present the transpose of this matrix. Because univariate Kriging assumes zero cross-covariances, this table contains the value 0. The corresponding elements in multivariate Kriging are virtually zero; e.g., -2.6E-14 in row 1 and column 2.

F Kriging for functions of outputs

Table F.1 shows $\overline{\widehat{\text{IMSE}^{(g)}}}$ for $y^{(1)}$, $y^{(2)}$, $y^{(3)} = y^{(1)} + y^{(2)}$, and $y^{(4)} = y^{(1)}y^{(2)}$. Table F.2 gives the t-statistics to test whether the RMLEs of the Kriging parameters significantly differ from the true values for $y^{(1)}$and $y^{(2)}$; for the outputs $y^{(3)}$and $y^{(4)}$ we do not know the true parameters, so we cannot apply these tests.

Table E.1: $\boldsymbol{\Sigma_{0;m;n}\Sigma_Y^{-1}}$ in multivariate and univariate Kriging

Multivar. $\rho = 0.8$		Multivar. $\rho = 0.95$		Univar. $\rho = 0.8$	
0.348906	-2.6E-14	0.348906	-2.7E-13	0.348906	0
-3.3E-16	0.348906	-9.4E-15	0.348906	0	0.348906
0.954562	1.6E-13	0.954562	0	0.954562	0
-4.9E-15	0.954562	-1.6E-14	0.954562	0	0.954562
-0.51635	-1.3E-13	-0.51635	4.55E-13	-0.51635	0
7.11E-15	-0.51635	7.13E-14	-0.51635	0	-0.51635
0.389122	-1.1E-13	0.389122	-2.3E-13	0.389122	0
-1E-14	0.389122	-1E-13	0.389122	0	0.389122
-0.29666	2.22E-13	-0.29666	-1.6E-12	-0.29666	0
1.06E-14	-0.29666	2.85E-14	-0.29666	0	-0.29666
0.216743	1.99E-13	0.216743	-6.8E-13	0.216743	0
-1.2E-14	0.216743	1.24E-14	0.216743	0	0.216743
-0.14675	1.11E-13	-0.14675	-4.8E-13	-0.14675	0
7.77E-15	-0.14675	-1.1E-14	-0.14675	0	-0.14675
0.087582	6E-14	0.087582	-2.1E-13	0.087582	0
-1.9E-15	0.087582	1.07E-14	0.087582	0	0.087582
-0.04163	-6.9E-15	-0.04163	1.14E-13	-0.04163	0
-1.3E-15	-0.04163	3.52E-15	-0.04163	0	-0.04163
0.012026	-1.1E-16	0.012026	-1.4E-14	0.012026	0
-1.2E-16	0.012026	2.21E-15	0.012026	0	0.012026

Table F.1: $\overline{\widehat{\text{IMSE}^{(g)}}}$ in univariate and multivariate Kriging

	Output 1		Output 2	
Experiment	Univar.	Multivar.	Univar.	Multivar.
1	0.000181	0.000915	0.0048008	0.011003
2	0.239103	0.245076	5.7627378	5.862312
3	0.000179	0.000748	0.0041866	0.011534
4	0.239072	0.246433	5.7128304	5.894214
	Output 3		Output 4	
Experiment	Univar.	Multivar.	Univar.	Multivar.
1	0.006536	0.016618	0.290038	0.241913
2	7.880984	8.020599	26.83617	28.45734
3	0.004553	0.01359	0.14392	0.152387
4	6.403224	6.576276	16.85455	18.1546

Table F.2: t-tests for RMLE in univariate and multivariate Kriging; four outputs

Experiment	μ_1	μ_2	θ_1	θ_2	σ_1^2	σ_2^2	σ_{12}
			Univariate Kriging				
1	1.151	1.152	2.026*	2.401*	0.567	-0.896	-
2	1.449	1.455	-6.388*	-4.935*	-0.369	-0.154	-
3	1.150	0.490	1.984	2.933*	0.565	-2.566*	-
4	1.450	0.771	-6.386*	-4.167*	-0.382	-1.530	-
			Multivariate Kriging				
Experiment	μ_1	μ_2	θ_1	θ_2	σ_1^2	σ_2^2	σ_{12}
1	0.856	1.000	9.014*	3.708*	1.307	3.524*	1.564
2	1.453	1.106	-16.804*	-15.662*	3.392*	6.285*	2.344*
3	0.855	0.770	5.805*	4.057*	3.474*	3.471*	2.246*
4	1.735	0.462	-19.102*	-17.015*	5.994*	5.118*	0.835

References

Álvarez, M. A., & Lawrence, N. D. (2011). Computationally efficient convolved multiple output Gaussian processes. *Journal of Machine Learning Research*, *12*, 1459–1500.

Álvarez, M. A., Rosasco, L., & Lawrence, N. D. (2011). Kernels for vector-valued functions: A review. *Foundations and Trends in Machine Learning*, *4*, 195–266.

Bonilla, E. V., Chai, K. M., & Williams, C. (2007). Multi-task Gaussian process prediction. In J. Platt, D. Koller, Y. Singer, & S. Roweis (Eds.), *Advances in Neural Information Processing Systems 20* (pp. 153–160). Cambridge, MA: MIT Press.

Box, G., Jenkins, G., & Reinsel, G. (1994). *Time series analysis: forecasting and control*. Forecasting and Control Series. Prentice Hall.

Boyle, P., & Frean, M. R. (2005). Dependent Gaussian processes. In L. K.

Saul, Y. Weiss, & L. Bottou (Eds.), *Advances in Neural Information Processing Systems 17* (pp. 217–224). Cambridge, MA: MIT Press.

Chen, X., Ankenman, B. E., & Nelson, B. L. (2012). The effects of common random numbers on stochastic Kriging metamodels. *ACM Transactions on Modeling and Computer Simulation*, *22*, 7:1–7:20.

Chen, X., Ankenman, B. E., & Nelson, B. L. (2013). Enhancing stochastic Kriging metamodels with gradient estimators. *Operations Research*, *61*, 512–528.

Constantinescu, E. M., & Anitescu, M. (2013). Physics-based covariance models for Gaussian processes with multiple outputs. *International Journal for Uncertainty Quantification*, *3*, 47–71.

Cressie, N. (1991). *Statistics for spatial data*. New York: Wiley.

Den Hertog, D., Kleijnen, J. P. C., & Siem, A. Y. D. (2006). The correct Kriging variance estimated by bootstrapping. *Journal of The Operational Research Society*, *57*, 400–409.

Forrester, A. I. J. (2010). Black-box calibration for complex-system simulation. *Philosophical Transactions of The Royal Society A: Mathematical, Physical and Engineering Sciences*, *368*, 3567–3579.

Forrester, A. I. J., Sobester, A., & Keane, A. J. (2008). *Engineering Design via Surrogate Modelling - A Practical Guide*. Chichester, U.K: Wiley.

Frazier, P. (2010). Wiley encyclopedia of operations research and management science. chapter Learning with Dynamic Programming. (pp. 1–13). New York: Wiley. In press.

Fricker, T., Oakley, J. E., & Urban, N. M. (2010). *Multivariate emulators with nonseparable covariance structures*. Technical Report Managing Uncertainty in Complex Models (MUCM).

Fricker, T. E., Oakley, J. E., & Urban, N. M. (2013). Multivariate Gaussian process emulators with nonseparable covariance structures. *Technometrics*, *55*, 47–56.

Gano, S. E., Renaud, J. E., Martin, J. D., & Simpson, T. W. (2006). Update strategies for Kriging models used in variable fidelity optimization. *Structural and Multidisciplinary Optimization*, *32*, 287–298.

Gneiting, T., Kleiber, W., & Schlather, M. (2010). Matern cross-covariance functions for multivariate random fields. *Journal of the American Statististics Association*, *105*, 1167–1177.

Goh, J., Bingham, D., Holloway, J. P., Grosskopf, M. J., Kuranz, C. C., & Rutter, E. (2013). Prediction and computer model calibration using outputs from multi-fidelity simulators. *Technometrics*, *55*, 501–512.

Hernandez, A. F., & Grover, M. A. (2013). Error estimation properties of Gaussian process models in stochastic simulations. *European Journal of Operational Research*, *228*, 131 – 140.

Higdon, D. (2002). Space and space-time modeling using process convolutions. In C. W. Anderson, V. Barnett, P. C. Chatwin, & A. H. El-Shaarawi (Eds.), *Quantitative Methods for Current Environmental Issues* (pp. 37–56). Springer.

Higdon, D., Gattiker, J., Williams, B., & Rightley, M. (2008). Computer model calibration using high-dimensional output. *Journal of the American Statististics Association*, *103*, 570–583.

Jones, D. R., Schonlau, M., & Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, *13*, 455–492.

Kleijnen, J. P. C. (1993). Simulation and optimization in production planning a case study. *Decision Support Systems*, *9*, 269–280.

Kleijnen, J. P. C. (2008). *Design and analysis of simulation experiments*. Springer-Verlag.

Kleijnen, J. P. C., Van Beers, W., & Van Nieuwenhuyse, I. (2010). Constrained optimization in simulation: a novel approach. *European Journal of Operational Research*, *202*, 164–174.

Kleijnen, J. P. C., Van Ham, G., & Rotmans, J. (1992). Techniques for sensitivity analysis of simulation models: A case study of the CO_2 greenhouse effect. *SIMULATION*, *58*, 410–417.

Kleijnen, J. P. C., & Mehdad, E. (2013). Parametric bootstrapping versus conditional simulation for the Kriging predictor variance and efficient global optimization. In *Simulation Conference (WSC), Proceedings of the 2013 Winter* (pp. 1–12).

Kleijnen, J. P. C., & Smits, M. T. (2003). Performance metrics in supply chain management. *Journal of Operational Research Society*, *54*, 507–514.

Li, G., Azarm, S., Farhang-Mehr, A., & Diaz, A. R. (2006). Approximation of multiresponse deterministic engineering simulations: a dependent metamodeling approach. *Structural and Multidisciplinary Optimization*, *31*, 260–269.

Li, R., & Sudjianto, A. (2005). Analysis of computer experiments using penalized likelihood in Gaussian Kriging models. *Technometrics*, *47*, 111–120.

Loeppky, J. L., Sacks, J., & Welch, W. J. (2009). Choosing the sample size of a computer experiment: A practical guide. *Technometrics*, *51*, 366–376.

Lophaven, S., Nielsen, H., & Sondergaard, J. (2002). *DACE: a MATLAB Kriging toolbox, version 2.0*. IMM Technical University of Denmark Lyngby, Denmark.

Mahevas, S., & Pelletier, D. (2004). Isis-fish, a generic and spatially explicit simulation tool for evaluating the impact of management measures on fisheries dynamics. *Ecological Modelling*, *171*, 65 – 84.

Marrel, A., Iooss, B., da Veiga, S., & Ribatet, M. (2010). Global sensitivity analysis of stochastic computer models with joint metamodels. *Statistics and Computing*, *22*, 833–847.

Press, W., Teukolsky, S., Vetterling, W., & Flannery, B. (1992). *Numerical Recipesin C*. The Art of Scientific Computing. Cambridge University Press.

Rasmussen, C. E., & Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning series)*. The MIT Press.

Roustant, O., Ginsbourger, D., & Deville, Y. (2012). DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by Kriging-based metamodeling and optimization. *Journal of Statistical Software*, *51*, 1–55.

Santner, T. J., Williams, B. J., & Notz, W. I. (2003). *The Design and Analysis of Computer Experiments*. New York: Springer-Verlag.

Seber, G., & Wild, C. (2003). *Nonlinear Regression*. Wiley Series in Probability and Statistics. Wiley-Interscience.

Simpson, T., Poplinski, J., Koch, P. N., & Allen, J. (2001). Metamodels for computer-based engineering design: Survey and recommendations. *Engineering with Computers*, *17*, 129–150.

Svenson, J. (2011). *Computer experiments: multiobjective optimization and sensitivity analysis*. Ph.D. thesis The Ohio State University The Ohio State University.

Svenson, J., & Santner, T. (2010). *Multiobjective optimization of expensive black-box functions via Expected Maximin Improvement*. Technical Report 43210 Ohio University Columbus, Ohio.

Tuo, R., Wu, C. F. J., & Yu, D. (2013). Surrogate modeling of computer experiments with different mesh densities. *Technometrics*, . Accepted.

Ver Hoef, J. M., & Barry, R. P. (1998). Constructing and fitting models for cokriging and multivariable spatial prediction. *Journal of Statistical Planning and Inference*, *69*, 275–294.

Wackernagel, H. (2003). *Multivariate geostatistics: an introduction with applications*. New York: Springer-Verlag.

Williams, B., Santner, T., Notz, W., & Lehman, J. (2010). Statistical modelling and regression structures - festschrift in the honour of ludwig fahrmeir. chapter Sequential design of computer experiments for constrained optimization. (pp. 449–472). Springer-Verlag.

Zhang, Z. (2007). *New modeling procedures for functional data in computer experiments*. Ph.D. thesis The Pennsylvania State University.